Kalman filter vs Complementary filter

Note: At the bottom of the post the complete source code

The use of accelerometer and gyroscope to build little robots, such as the self-balancing, requires a math filter in order to merge the signals returned by the sensors.
The gyroscope has a drift and in a few time the values returned are completely wrong. The accelerometer, from the other side, returns a true value when the acceleration is progressive but it suffers much the vibrations, returning values of the angle wrong.

Usually a math filter is used to mix and merge the two values, in order to have a correct value: the Kalman filter . This is the best filter you can use, even from a theoretical point of view, since it is one that minimizes the errors from the true signal value. However it is very difficult (see here) to understand. In fact, the filter needs to be able to calculate the coefficients of the matrices, the process-based error, measurement error, etc. that are not trivial.

In the hobbistic world, recently are emerging other filters, called **complementary filters**. In fact, they manage both high-pass and low-pass filters simultaneously. The low pass filter filters high frequency signals (such as the accelerometer in the case of vibration) and low pass filters that filter low frequency signals (such as the drift of the gyroscope). By combining these filters, you get a good signal, without the complications of the Kalman filter.

Making a study from a theoretical point of view, the discussion is complicated and is beyond the scope of this tutorial. The complementary filters can be have different 'orders'. Here I speak about the so-called first-order filter that filter already well, and the second-order filter which filter even better. Clearly, going from first to second order, the algorithm is more complicated to use and perhaps there is no gain so obvious to justify the increase in complexity.
A great introduction to the first order complementary filters applied to the an accelerometer and a gyroscope, comes from MIT (here). It introduces the filter in a a very simple mode. On this document is based the first Arduino algorithm:

```
// a=tau / (tau + loop time)
// newAngle = angle measured with atan2 using the accelerometer
// newRate = angle measured using the gyro
// looptime = loop time in millis()

float tau=0.075;
float a=0.0;

float Complementary(float newAngle, float newRate,int looptime) {
float dtC = float(looptime)/1000.0;
```

VISITOR MAP

Large Visitor Map

ARCHIVE

- March 2016
- October 2014
- June 2013
- May 2013
- December 2012
- May 2012
- April 2012
- March 2012
- December 2011
- October 2011
- September 2011
- August 2011
- July 2011
- June 2011

CATEGORIES

- Auduino
- compound eye
- delta robot
- Dongbu Herkulex
- Dynamixel
- GRBL
- OpenCv
- robot brush
- robot drawing

```
a=tau/(tau+dtC);
x_angleC= a* (x_angleC + newRate * dtC) + (1-a) * (newAngle);
return x_angleC;
}
```

It ‘enough to choose the response time of *tau*, to send the arguments, ie the angle measured with the accelerometer and the gyroscope, the time of the loop and you get in two lines, the angle calculated by the filter.

The algorithm at the base of the second order complementary filter is described here. Indeed it is not described at all, but now we've figured out how the filter works by the MIT's documentation. The principle is the same, the algorithm is more complicated. The translation of this algorithm for the Arduino:

```
// newAngle = angle measured with atan2 using the accelerometer
// newRate = angle measured using the gyro
// looptime = loop time in millis()

float Complementary2(float newAngle, float newRate,int looptime) {
float k=10;
float dtc2=float(looptime)/1000.0;

x1 = (newAngle -   x_angle2C)*k*k;
y1 = dtc2*x1 + y1;
x2 = y1 + (newAngle -   x_angle2C)*2*k + newRate;
x_angle2C = dtc2*x2 + x_angle2C;

return x_angle2C;
}
```

Here too we just have to set the *k* and magically we get the angle.

If we want to apply the Kalman filter, we can re-use one of the codes already present in internet. This is the code that I copied from the Arduino forum (here):

```
// KasBot V1 - Kalman filter module

float Q_angle  =  0.01; //0.001
float Q_gyro   =  0.0003;  //0.003
float R_angle  =  0.01;  //0.03

float x_bias = 0;
float P_00 = 0, P_01 = 0, P_10 = 0, P_11 = 0;
float  y, S;
float K_0, K_1;

// newAngle = angle measured with atan2 using the accelerometer
// newRate = angle measured using the gyro
// looptime = loop time in millis()

float kalmanCalculate(float newAngle, float newRate,int looptime)
{
float dt = float(looptime)/1000;
x_angle += dt * (newRate - x_bias);
P_00 +=  - dt * (P_10 + P_01) + Q_angle * dt;
P_01 +=  - dt * P_11;
P_10 +=  - dt * P_11;
P_11 +=  + Q_gyro * dt;

y = newAngle - x_angle;
S = P_00 + R_angle;
K_0 = P_00 / S;
K_1 = P_10 / S;

x_angle +=  K_0 * y;
x_bias  +=  K_1 * y;
P_00 -= K_0 * P_00;
P_01 -= K_0 * P_01;
P_10 -= K_1 * P_00;
P_11 -= K_1 * P_01;

return x_angle;
```


- Robottini
- Senza categoria
- Speech
- Tips
- vacuum gripper

ROBOT BLOG

- Appiphania
- billporter
- Gioblu (in Italian)
- Patrick Mc Cabe Makes
- Savage electronics (in spanish)
- seriousrobotics

ROBOT SITES

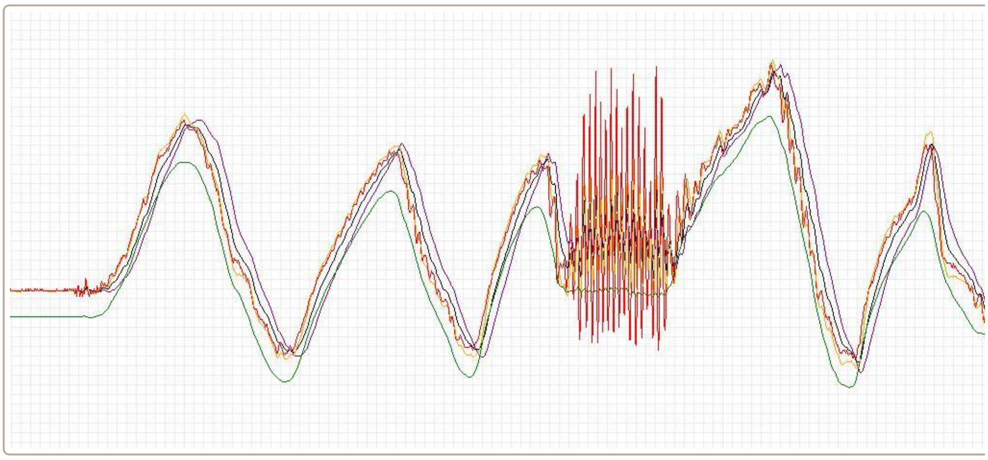
- JavaCvPro site
- letsmakerobots
- Privacy policy
- Robotics Portal
- societyofrobots

TAG

Arduino audio chip auduino complementary filter compound eye Delta Robot diffident robot dog robot Dongbu graphic head robot Herkulex IR sensor javacv JavaCvPro Kalman filter Line follower me motor OpenCV plot Processing quadrapod robot robottini Self-balancing Serial Port servo servos controller SOMO-14D tail robot Ultrasonic sensor vacuum gripper

```
}
```

To get the answer, you have to set 3 parameters: *Q_angle, R_angle,R_gyro*. The activity is a bit complicated .
But what happens with these algorithms? Similar curves are obtained? Here's a comparison:
There are 5 curves:

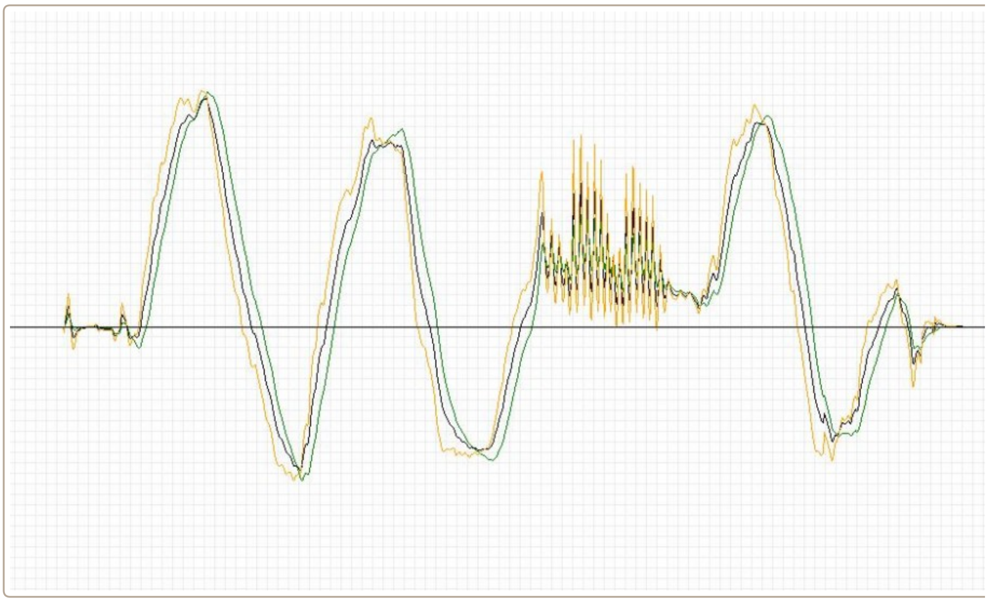


Color lines:

- Red – accelerometer
- Green – Gyro
- Blue – Kalman filter
- Black – complementary filter
- Yellow – the second order complementary filter

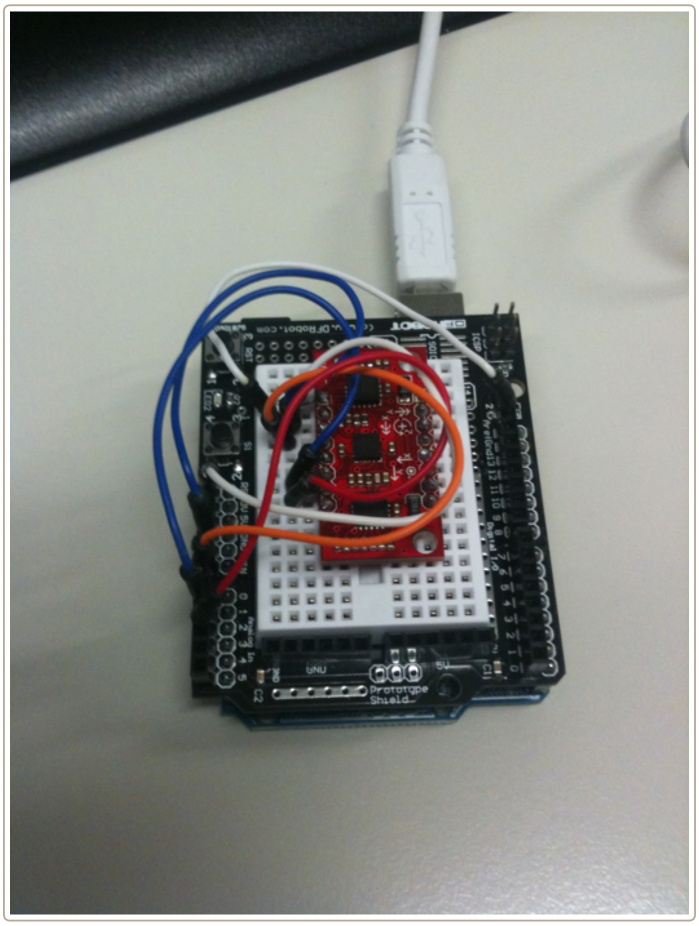
As you can see the signals filtered are very similarly. Note that in the presence of vibrations, the accelerometer (red) generally go crazy. The gyro (green) has a very strong drift increasing int the time.

Now let's see a comparison only between a filtered signal. That kalman (green), complementary (black) and complementary second-order (yellow). You can see how the Kalman is a bit late vs complementary filters, but it is more responsive to the vibration. In this case the second order filter does not return an ideal curve, probably I have to work a bit on the coefficients.



In conclusion I think that the complementary filter, in this case the first order, can be used in place of the Kalman filter. The smoothing is good and the algorithm is much simpler than Kalman.

The hardware I used was composed of:
– Arduino 2009
– 6-axis IMU SparkFun Razor 6 DOF



This is the complete source code: Filters1

Tags: complementary filter, Kalman filter

This entry was posted on 25 September 2011, 07:11 and is filed under Tips. You can follow any responses to this entry through RSS 2.0. You can leave a response, or trackback from your own site.

43 Comments (and 5 trackbacks)

*#1 by **Iwan** on 21 November 2012 - 05:40*

how do you display the graph the results of Kalman and complementary?
This chart comes from the hardware or the results of the simulation matlab?

Tell me please,

Regard,
Iwan

*#2 by **robottini** on 21 November 2012 - 06:02*

No Matlab simulation. They are the hardware results. The code is included in the post.

*#3 by **Iwan** on 3 December 2012 - 19:55*

can you give me a complete source code? I want to try it..

*#4 by **robottini** on 3 December 2012 - 21:27*

I sent you an email with the code

*#5 by **xw** on 6 December 2012 - 22:35*

can you give me a complete source code? I want to try it..

*#6 by **ducati** on 7 December 2012 - 00:54*

nice post!

I, too, am interested in how you collect the data for the graph.

if possible, please send me the code.

ciao e grazie mille!

*#7 by **tantun** on 7 December 2012 - 15:14*

can you send to me complete source code?

*#8 by **Izbert** on 11 December 2012 - 03:56*

Can i also get a copy of this source code?

*#9 by **phuongnut** on 12 December 2012 - 16:21*

I want to try it.
Send to me complete source code pls!

*#10 by **robottini** on 12 December 2012 - 17:50*

FOR ALL: I put the source code in the post. please take it if you need

*#11 by **jj** on 1 January 2013 - 11:34*

Are the codes that you post are all complete source codes? If not can I have them? Thanks

*#12 by **robottini** on 1 January 2013 - 13:03*

Yes, i put the code in the post.

*#13 by **kamal** on 19 January 2013 - 09:57*

Dear , can u send me to code

*#14 by **robottini** on 19 January 2013 - 10:16*

At the bottom of the post the complete source code

*#15 by **Someone** on 30 January 2013 - 05:47*

Can you send a completed code to me? I want to try it since I do not have any experience about that. Thanks

*#16 by **robottini** on 30 January 2013 - 08:39*

At the bottom of the post the complete source code

*#17 by **Someone** on 30 January 2013 - 09:13*

Okay, I can see that. Thanks

*#18 by **Morten** on 2 May 2013 - 18:25*

Hi, when I click on Filters1 for downloading the source code I just get redirected to this page.

Is it possible to repost the source code?

*#19 by **robottini** on 3 May 2013 - 07:48*

Thanks Morten, now the link works.

*#20 by **Pablopaolus** on 27 May 2013 - 16:29*

Hello,

Thank you for sharing your job 🙂

I'm using PIC18F46J50 as MCU along with Sparkfun IMU – 6DOF ITG3200/ADXL345, and I'm trying to combine accelerometer and giroscope data using the first order complementary filter. I've ported your code (to CCS v4.140). It seems to work well enough within an angle range. However, if I tilt for instance pitch angle to 90 or -90 deg, roll angle goes crazy. The same occurs to pitch if I tilt roll to +-90. I don't know if I've explained myself clearly, so I've made a video:

http://www.youtube.com/watch?v=RmqbKVueVIw

Here is my code:

```
float tau=0.075;
float a=0.0;
float rollAcc=0, rollGyro=0;
```

float pitchAcc=0, pitchGyro=0;

//RwAcc[x] in g
//Gyro[x] in deg/s

void ComplementaryFilter2() {
interval = (0.250*(float)timerCount250us)/1000.0; //Units: sec
timerCount250us = 0; //Restart the counters

rollAcc=atan2(RwAcc[1], RwAcc[2]) * 180 / PI;
pitchAcc=atan2(RwAcc[0], RwAcc[2]) * 180 / PI;

a=tau/(tau+interval);
roll = a * (roll + Gyro[0] * interval) + (1-a) * (rollAcc);
pitch = a * (pitch + Gyro[1] * interval) + (1-a) * (pitchAcc);
}

I use an internal interruption so as to calculate the interval. Timer0 overflows every 250us:

#int_TIMER0
void TIMER0_isr(void)
{
timerCount250us+=1;
set_timer0(64036);
}

Sorry for the long post.
I would greatly appreciate if you could help me.

Thank you.

*#21 by **Tjaart** on 10 June 2013 - 21:17*

Hello, I'm currently trying to implement a Kalman filter using the code above. After I plotted the accelerometer angle vs the Kalman angle, they seemed to be about the same. After looking through the code I found that :
x_angle += K_0 * y;
x_angle += K_0 * (newAngle – x_angle);
x_angle += (P_00 / S) * (newAngle – x_angle);
x_angle += (P_00 / (P_00 + R_angle)) * (newAngle – x_angle);

In most cases my P_00>>R_angle (sometimes maxed at about +-1800) whis basically means K_0~1 and then

x_angle = x_angle + newAngle – x_angle;
x_angle = newAngle;

Am I calculating P_00 wrong? Any help would be appreciated!

*#22 by **irina** on 13 June 2013 - 21:37*

Hi, I'm trying to use a ADXL345 and ITG 3200 for a platform with 3 analog servos. I need to filtrate the data from gyro and accel and using your complementary filter implementations. Hoe would the filter look like for all three axis active? thanks

Pingback: How to combine data from Gyro and Accel | solenerotech EN

Pingback: Come combinare i dati del giroscopio e dell'accelerometro | solenerotech IT

#23 by siddharth on 28 October 2013 - 12:43

Dear Sir,

plz send me source code of 2 wheel balancing robot using gyro-521 and kalman filter.
I will be very thankfulto you.

#24 by siddharth on 28 October 2013 - 12:44

Dear Sir,
plz send me the code of two wheel balancing robot using gyro and kalman filter.

*#25 by **robottini** on 28 October 2013 - 12:48*

Sorry the code is complex, I can't work on it. You can find many implementations on internet about it.

*#26 by **kevan** on 7 November 2013 - 14:55*

sir, can you help me. why result the source code can't look in serial monitor arduino?

*#27 by **robottini** on 7 November 2013 - 15:03*

Are you using the right speed in the serial monitor (the same of Serial.begin in the code)?

*#28 by **kevan** on 12 November 2013 - 09:13*

no sir, i'm using serial 1200.
it's make value of the sensor just 0?

*#29 by **robottini** on 12 November 2013 - 10:31*

You have to use the same speed in the serial monitor and in the Serial.begin in the code. Please use 57600 as a value in the Serial.beglin (57600) and in the Serial monitor. This is the first point. After, we can see if there are other problems.

*#30 by **kevan** on 12 November 2013 - 17:24*

sir, i was changed serial begin become 57600 but the result constant 0..
i'm using accelerometer (ADXL345) and Gyroscope (ITG3200), how setting and read this sensor sir?

*#31 by **Eric** on 23 December 2013 - 10:02*

我想回覆#20 by Pablopaolus
因為當在計算尤拉角的時候，當pitch在+-90度時，轉換矩陣會有奇異點的發生。

Here is the equation:
http://ppt.cc/oY1M
http://ppt.cc/7sc~

thx~ 😀

#32 by **Ali Hamza** on 2 February 2014 - 17:21

Hi, I was looking at your code and I kinda get hold of it and understand it. However, I have a question. This code gives all the 6 axis readings. like the 3 axis of gyro and 3 axis of accelerometer. Because I am trying to implement it on my quadroter. So wanted to know if I can implement it on it.

Sorry if I dnt make too much sense as I am new to all this.

#33 by **TARIKU w/TSADIK** on 23 March 2014 - 15:44

Hi, I am tried to implement Kalman filter for noisey Gyro-accelerometer data in matlab. Is there anyone who could help me ,please?

Pingback: Still working on the Autopilot | m/v C:[ESC]

#34 by **hari** on 14 May 2014 - 13:56

Hi Bro:)
Good job,
I need your guidance to build my quadcopter with kalman filter.
please,send your code…
thanks in advance

#35 by **Animesh** on 16 June 2014 - 19:50

I tried this code on a MPU 6050 GY 521 Breakboard..When I run the code I get all values as 0..
Can U suggest me the modifications that will be required or mail me the code??

#36 by **robottini** on 16 June 2014 - 20:48

Sorry, I don't know the MPU6050, but you can find many tutorials with google

#37 by **Elijah** on 30 June 2014 - 11:52

Hello,

Can someone help me with the code to implement the Kalmer and complementary filter?
Thanks.
Regards.

#38 by **Boston magician** on 6 September 2014 - 11:22

I savor, result iin I discovered exactly what I was taking a look
for. You have ended my fokur day long hunt! God Bless yyou man. Have a nice day.
Bye

Pingback: Gyro Drift And What To Do About It? | paulcreaser

Pingback: Gyro Drift And What To Do About It? | paulcreaser

*#39 by **Yang** on 2 October 2015 - 23:06*

I think there may be a big problem with deriving angle/heading by accelerometer data (which is used as the z- measurements in this Kalman example).

it accumulates current angle by integrating angle changes, which angle change is derived from v_x , v_y changes. but if you just spin around the exact point of the accelerometer, you get dv_x = 0 and dv_y = 0, but apparently the angle change is not zero

*#40 by **ddd** on 14 December 2015 - 15:10*

Thanks for information. I want to apply this filters to one dimensional data like temperature or pressure. How can i apply this.

*#41 by **Jim Remington** on 31 January 2016 - 05:45*

Although this post is rather old, continued interest in it compels me to point out a problem. The Kalman filter used here for comparison seems to work, but it is simply wrong. An explanation for several of its problems and an alternative that works well is presented at
http://home.earthlink.net/~schultdw/bbot/bbot.html

For anyone interested to learn the theory, I recommend an excellent text book entitled “Optimal State Estimation” by Dan Simon. Note: I have no connection with any of the authors.

*#42 by **naga** on 25 February 2016 - 14:05*

i need filter codes for gy85
‘

*#43 by **Richard Payne** on 12 May 2016 - 20:46*

I would like this code to try out, much appericated.

Your name (required)

Your e-mail (required, will not be published)

Your website

Type your comment here

Submit Comment